Metric LP Relaxation

Anil Maheshwari

anil@scs.carleton.ca
School of Computer Science
Carleton University
Canada

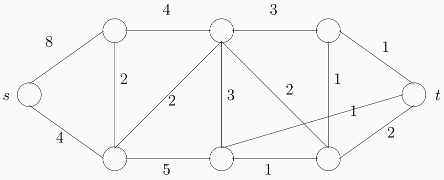
Outline

Min Cost st-cut

Min Cost $s-t$ cut in a Graph

Input: An undirected graph $G=(V,E)$ on n vertices and each edge has a positive weight $w: E \rightarrow \Re^{+}$. It will be easier to think of G as a complete graph K_n, as all the edges in $K_n \setminus G$ are assigned a weight of 0. Two specific vertices s and t of G.

Output: Find a set of edges $C \subseteq E$ of minimum total weight so that the graph $G'=(V, E \setminus C)$ has no path that between s and t. I.e., C forms a cut of minimum weight that separates s and t.



Towards LP Formulation

- Assume C is a cut.
- Define an indicator variable x_e for each edge e as follows:

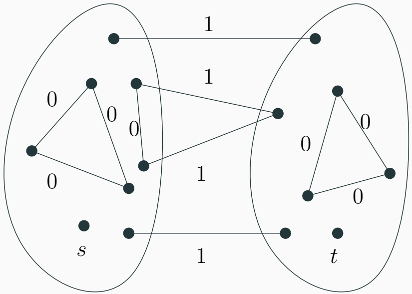
$$x_e = \begin{cases} 1, & \text{if } e \in C, \\ 0, & \text{otherwise} \end{cases}$$

Observations:

1. Cost of Cut equals $\sum_{e \in E} x_e w_e$.
2. Length of any path $\pi(s,t)$ joining s and t is ≥ 1. Length of π is defined as the sum total of x_e's values of the edges of π.

Metric Property

The variables x_e's assigned to the edges of G satisfy the metric property.



Linear Programming Formulation

Problem: Given a complete graph $G = (V, E)$, where edges have non-negative weights $e : E \to \Re^+$, and two vertices s and t, find the cut of minimum total weight that separates s and t.

(Integer) Metric LP Formulation

$$\min \sum_{e \in E} w_e x_e$$

Subject to:

1. Membership in the Cut: For each edge $e \in E$, $x_e \in \{0, 1\}$
2. Cut Constraint: $x_{st} \geq 1$.
3. Triangle Inequality: For every set of three distinct vertices $u, v, w \in V$: $x_{uw} + x_{wv} \geq x_{uv}$

Relaxed LP

Replace the constraint $x_e \in \{0, 1\}$ by $0 \leq x_e \leq 1$.

Relaxed Metric LP

$$\min \sum_{e \in E} w_e x_e$$

Subject to:

1. For each edge $e \in E$, $0 \leq x_e \leq 1$
2. $x_{st} \geq 1$.
3. For every set of three distinct vertices $u, v, w \in V$: $x_{uw} + x_{wv} \geq x_{uv}$

After solving the Relaxed LP, let $x_e \in [0,1]$ be the assignment of x values to each edge $e \in E$, and let $z^* = \sum_{e \in E} w_e x_e$ be the value of the objective function.

Note that x_e values satisfy:

1. Triangle Inequality
2. For any path in G between s and t, the length of the path is ≥ 1.
3. The cost of an optimal min cut in G is at least z^*.

Obtaining a Cut from Relaxed LP

Method to find the edges in the cut:

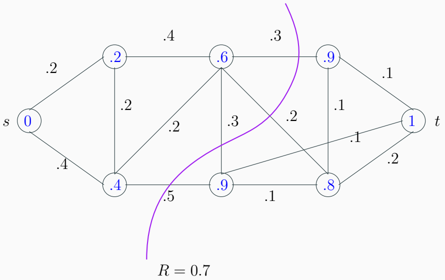
Step 1: Solve the Relaxed Metric LP to obtain x_e values for each edge $e \in E$.

Step 2: For each vertex $v \in V$, find the shortest distance $\delta(s, v)$ from s with respect to x_e values on edges.

Step 3: Choose an arbitrary value $R \in (0, 1)$.

Step 4: For each edge $e = (uv) \in E$ (assume $\delta(s, u) \leq \delta(s, v)$), place e in the cut if $\delta(s, u) < R < \delta(s, v)$.

Step 5: Return the edges in the cut.



Cost of the Cut

Claim

The expected sum total of the weights of the edges in the cut is at most z^*.

Proof: Let C be the collection of edges in the cut with respect to $R \in (0,1)$.
Consider an arbitrary edge $e = (uv) \in E$.
What is the probability that $e \in C$?

$e \in C$ if $\delta(s,u) < R < \delta(s,v)$, i.e. $R \in (\delta(s,u), \delta(s,v))$
Therefore, $Pr(e \in C) = \frac{\delta(s,v)-\delta(s,u)}{1} = \delta(s,v) - \delta(s,u)$.
Because of the triangle inequality $\delta(s,v) - \delta(s,u) \leq x_e$

Thus, $Pr(e \in C) \leq x_e$.

$E[cost(C)] = \sum_{e \in E} w_e Pr(e \in C) \leq \sum_{e \in E} w_e x_e = z^*$. $\square$

Finding an optimal Cut

- Notice that when R ranges from 0 to 1, one by one vertices are added to the component containing s.
- In all there are $n = |V|$ such events
- We can find all the events and return the cut that minimizes the total weight.

Observe:

1. If for some R the cost of the cut is $> z^*$, than there must be a cut for which the cost $< z^*$, since the average (i.e. expected) value is z^*
2. The cost of any cut can't be smaller than z^* (as z^* is the objective value of relaxed LP) $\implies$ the cut returned by the method is of optimal cost for any $R \in (0, 1)$

Theorem

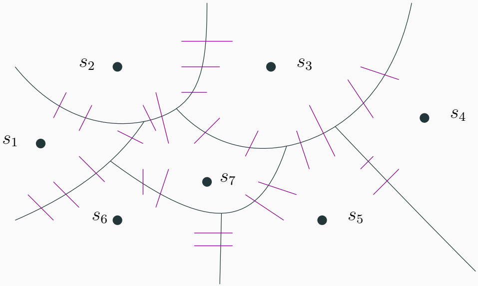
We can find an optimal cut in polynomial time using the Metric LP relaxation.

Multiway Min Cut

Multiway Cuts

Input: An undirected (complete) graph $G = (V, E)$ on n vertices and each edge has a positive weight $w : E \to \Re^+$. A set $T = \{s_1, \ldots, s_k\} \subset V$ of k vertices called terminals.

Output: Find a set of edges $C \subseteq E$ of minimum total weight so that the graph $G' = (V, E \setminus C)$ has no path between any pair of terminals in T.



Triangle Inequality

Edges in the Cut

Let C be a multiway cut that separates every pair of terminals. Define an indicator variable x_e for each edge e as follows:

$$x_e = \begin{cases} 1, & \text{if } e \in C, \\ 0, & \text{otherwise} \end{cases}$$

The assignment of x_e values to each edge satisfies:

1. Cost of Multiway Cut equals $\sum_{e \in E} x_e w_e$.
2. For any pair of distinct terminals $s_i, s_j \in T$, the length of any path $\pi(s_i, s_j)$ joining s_i and s_j is ≥ 1.
3. x_e values satisfy the triangle inequality. I.e., for any three distinct vertices $u, v, w \in V$, $x_{uw} + x_{wv} \geq x_{uv}$.

Linear Programming Formulation for Multiway Cuts

Problem: Given a complete graph $G = (V, E)$, where edges have non-negative weights $e : E \to \Re^+$, and a set $T \subset V$ of k terminals, find the cut of minimum total cost that separates every pair of terminals.

(Integer) Metric LP Formulation

$$\min \sum_{e \in E} w_e x_e$$

Subject to:

1. Membership in the Cut: For each edge $e \in E$, $x_e \in \{0, 1\}$
2. Cut Constraint: For every distinct pair $s_i, s_j \in T$, $x_{s_i s_j} \geq 1$.
3. Triangle Inequality: For every set of three distinct vertices $u, v, w \in V$: $x_{uw} + x_{wv} \geq x_{uv}$

Relaxed LP

Replace the constraint $x_e \in \{0, 1\}$ by $0 \leq x_e \leq 1$.

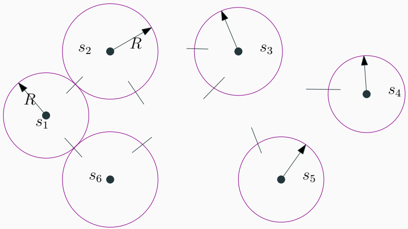
Step 1: $C \leftarrow \emptyset$

Step 2: Solve the Relaxed Metric LP to obtain x_e values for each edge $e \in E$.

Step 3: Choose an arbitrary value $R \in (0, 1/2)$.

Step 4: For each vertex $s_i \in T$, find the shortest distances $\delta(s_i, v)$ from s_i with respect to x_e values on edges. For each edge $e = (uv) \in E$ (assume $\delta(s_i, u) \leq \delta(s_i, v)$), place e in the cut C if $\delta(s_i, u) < R < \delta(s_i, v)$.

Step 5: Return the set of edges C.



Non-overlapping Balls

Let $s_i \in T$ be a terminal, and let $R \in (0, 1/2)$.
Define $B(s_i, R) = \{v \in V | \delta(s_i, v) < R\}$.
$B(s_i, R)$ consists of all the vertices that are within the distance R of s_i.

Disjointness of $B(s_i, R)$ and $B(s_j, R)$

Let $s_i, s_j \in T$ be two distinct terminals and let $B(s_i, R)$ and $B(s_j, R)$ be the set of vertices within distance of $R \in (0, 1/2)$ of s_i and s_j, respectively.
Then, $B(s_i, R) \cap B(s_j, R) = \emptyset$

Observation 1

Consider any edge $e = (uv) \in E$, where $u \in B(s_i, R)$. The edge $e \in C$ if $v \notin B(s_i, R)$.

Feasibility

Observation 2

The cut C returned by the method is a feasible multiway cut.

Proof: We need to show that there is no path between any pair of distinct terminals $s_i, s_j \in T$ in the graph $G - C$.

By the 2nd constraint of LP, $x_{s_i s_j} \geq 1$.

From the triangle inequality any path $\pi(s_i, s_j)$ between s_i and s_j in G will have length ≥ 1. Alternatively, for any vertex $w \in V$, $\delta(s_i, w) + \delta(s_j, w) \geq 1$.

Distance between any two vertices in a ball $B(s_i, R)$ is < 1.

Thus, for any ball $B(s_i, R)$, only terminal that is in $B(s_i, R)$ is s_i, i.e., $B(s_i, R) \cap T = s_i$.

Hence each connected component of $G \setminus C$ contains at most one terminal.

□

Bounding Probability of an Edge to be in C

Observation 3

Let $e = (u, v)$ be an edge in G. $Pr(e \in C) \leq 2x_e$.

Proof: Define sets $X_1, \ldots, X_k$, where $X_i = \{v \in V | \delta(s_i, v) < 1/2\}$.
Note that for any pair of distinct sets X_i, X_j, $X_i \cap X_j = \emptyset$ and $B(s_i, R) \subseteq X_i$.

For the edge $e = (u, v)$, one of the following cases occurs

Case 1: None of the endpoints u, v are in any set.
($\implies e \notin C$ and $Pr(e \in C) = 0 \leq 2x_e$)

Case 2: Both u, v are in the same set, say X_i.

Case 3: $u \in X_i$ and $v \in V \setminus X_i$.

We need to estimate $Pr(e \in C)$ for Cases 2 and 3.

Probability of an Edge to be in C (contd.)

Case 2: $u, v \in X_i$. Assume $\delta(s_i, u) \leq \delta(s_i, v)$.
We know $R \in (0, 1/2)$.
$e = (u, v)$ will be in the cut C if $\delta(s_i, u) < R$ and $\delta(s_i, v) > R$.
By triangle inequality we know that $\delta(s_i, v) - \delta(s_i, u) \leq x_e$.
Since we are choosing R uniformly at random in $(0, 1/2)$,
$Pr(e \in C) = \frac{\delta(s_i,v)-\delta(s_i,u)}{\frac{1}{2}} \leq 2x_e$.

Case 3: $u \in X_i$ and $v \in V \setminus X_i$.
We know $\delta(s_i, u) < 1/2$ and $\delta(s_i, v) \geq 1/2$.
By triangle inequality we know that $\delta(s_i, v) - \delta(s_i, u) \leq x_e$.
$e = (u, v)$ will be in the cut C if $\delta(s_i, u) < R$.
$Pr(e \in C) = Pr(R \in (\delta(s_i, u), 1/2)) \leq \frac{\frac{1}{2}-\delta(s_i,u)}{\frac{1}{2}} \leq 2(\frac{1}{2} - \delta(s_i, u)) \leq$
$2(\delta(s_i, v) - \delta(s_i, u)) \leq 2x_e$.

□

Probability of an Edge to be in C (contd.)

Case 2: $u, v \in X_i$. Assume $\delta(s_i, u) \leq \delta(s_i, v)$.
We know $R \in (0, 1/2)$.
$e = (u, v)$ will be in the cut C if $\delta(s_i, u) < R$ and $\delta(s_i, v) > R$.
By triangle inequality we know that $\delta(s_i, v) - \delta(s_i, u) \leq x_e$.
Since we are choosing R uniformly at random in $(0, 1/2)$,
$Pr(e \in C) = \frac{\delta(s_i,v) - \delta(s_i,u)}{\frac{1}{2}} \leq 2x_e$.

Case 3: $u \in X_i$ and $v \in V \setminus X_i$.
We know $\delta(s_i, u) < 1/2$ and $\delta(s_i, v) \geq 1/2$.
By triangle inequality we know that $\delta(s_i, v) - \delta(s_i, u) \leq x_e$.
$e = (u, v)$ will be in the cut C if $\delta(s_i, u) < R$.
$Pr(e \in C) = Pr(R \in (\delta(s_i, u), 1/2)) \leq \frac{\frac{1}{2} - \delta(s_i,u)}{\frac{1}{2}} \leq 2(\frac{1}{2} - \delta(s_i, u)) \leq 2(\delta(s_i, v) - \delta(s_i, u)) \leq 2x_e$.

□

Note: If $u \in X_i$ and $v \in X_j$, then part of e lies in $B(s_i, 1/2)$ and part in $B(s_j, 1/2)$.
Observe that $(1/2 - \delta(s_i, u)) + (1/2 - \delta(s_j, v)) \leq x_e$.

$Pr(e \in C) \leq 2((1/2 - \delta(s_i, u)) + (1/2 - \delta(s_i, u))) \leq 2x_e$.

Expected Cost of the Cut

Observation 4

The expected weight of the edges in the multiway cut is at most $2z^*$, where z^* is the value of the objective function returned by the LP relaxation ($z^* = \sum_{e \in E} w_e x_e$).

Proof: Let C be the collection of edges in the cut with respect to $R \in (0, 1/2)$.

We have already seen that for an arbitrary edge $e \in E$, $Pr(e \in C) \leq 2x_e$.

$$\begin{aligned} E[cost(C)] &= \sum_{e \in E} w_e Pr(e \in C) \\ &\leq \sum_{e \in E} w_e \times 2x_e \\ &= 2 \sum_{e \in E} w_e x_e \\ &= 2z^* \qquad \square \end{aligned}$$

2-approximation of Multiway cut

Theorem

Let $G = (V, E)$ be a simple (complete) graph where each edge has a non-negative real weight. Let $T \subset V$ be a set of terminals. We can find a set $C \subseteq E$ with the following properties:

1. $G - C$ has no path connecting any pair of terminals.
2. The total weight of the edges in C is at most 2 times the weight of an optimal multiway cut.
3. We can determine C in polynomial time using the solution of the relaxed LP.

Integrality Gap

Consider an unweighted star graph with $k+1$ vertices. It consists of k-leaves and all of them are connected to a central node.
Let the k leaves constitute the set T of terminals.

Cost of Optimal solution $= k-1$ (remove any set of $k-1$ edges.)

Cost of relaxed LP is $k/2$ (set cost of each edge to $1/2$).

Approximation Factor $= \frac{k-1}{\frac{k}{2}} = 2(1-\frac{1}{k})$

Using this approach, we can't do better in the worst case (termed as the integrality gap).

Remark: A different LP relaxation yields a $\frac{3}{2}$-approximation.

Multicuts in Graphs

Multicuts in General Graphs

Input: A (complete) graph $G = (V, E)$ with non-negative weights on edges and a set of k-vertex pairs $(s_1, t_1), (s_2, t_2), \ldots, (s_k, t_k)$.

Output: A set of edges $C \subseteq E$ of minimum total weight so that $G \setminus C$ has no path between s_i and t_i for $i = 1, \ldots, k$.

Edges in the Cut

Let C be a multiway cut that separates every pair $(s_1, t_1), (s_2, t_2), \ldots, (s_k, t_k)$. Define an indicator variable x_e for each edge e as follows:

$$x_e = \begin{cases} 1, & \text{if } e \in C, \\ 0, & \text{otherwise} \end{cases}$$

The assignment of x_e values to each edge satisfies:

1. Cost of Multicut equals $\sum\limits_{e \in E} x_e w_e$.
2. For any pair (s_i, t_i), length of any path between them is ≥ 1, where the length of an edge e is its x_e value.
3. For any three distinct vertices u, v, w, $x_{uw} + x_{wv} \geq x_{uv}$.

Problem: Given a (complete) graph $G = (V, E)$, where edges have non-negative weights $e : E \to \Re^+$, and k pairs $(s_1, t_1), (s_2, t_2), \ldots, (s_k, t_k)$, find the cut of minimum total weight that separates vertices in each pair.

(Integer) Metric LP Formulation

$$\min \sum_{e \in E} w_e x_e$$

Subject to:

1. Membership in the Cut: For each edge $e \in E$, $x_e \in \{0, 1\}$
2. Cut Constraint: For every pair (s_i, t_i), $x_{s_i t_i} \geq 1$.
3. Triangle inequality: For any three distinct vertices u, v, w, $x_{uw} + x_{wv} \geq x_{uv}$.

Relaxed LP

Replace the constraint $x_e \in \{0, 1\}$ by $0 \leq x_e \leq 1$.

Algorithm for finding the edges in the cut

Initialize:

1. Choose an $R \in (0, 1/2)$, uniformly at random. Initialize the cut $C \leftarrow \emptyset$.
2. Define k blocks $X_1 = \ldots = X_k = \emptyset$
3. Unmark all the vertices in G.

Main Steps:

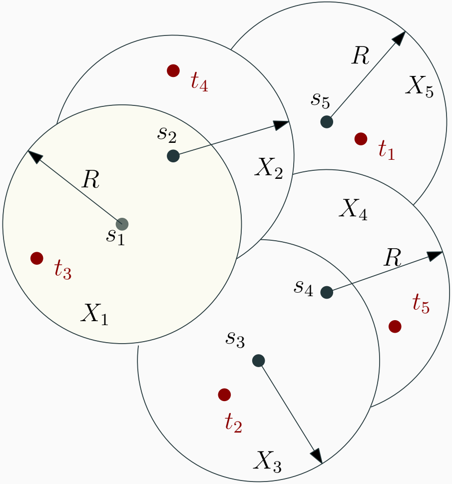
Step 1: Compute a random permutation of vertices $s_1, s_2, \ldots, s_k$.
WLOG, assume the ordering is $s_1, s_2, \ldots, s_k$.

Step 2: Let $B_i(s_i, R)$ be the ball consisting of all the vertices within distance R of s_i.
For each s_i in the order of permutation do:
For each unmarked vertex $v \in B(s_i, R)$, mark v and place it in the block X_i.

Step 2: For each edge $e = (u, v) \in E$, place it in the cut C if $u \in X_\alpha$ and $v \notin X_\alpha$ for some $\alpha \neq \beta$.

Step 3: Return C.

An Illustration



Observations

Observation 1

$$X_i = B(s_i, R) \setminus \bigcup_{j=1}^{i-1} B(s_j, R).$$

Observation 2

For each pair s_i, t_i, $i = 1, \ldots, k$, the following holds:

1. $t_i \notin X_i$
2. if $s_i \in X_j$ then $t_i \notin X_j$.

Proof: Since $R < 1/2$ and $x_{s_i t_i} \geq 1$ (by LP), $t_i \notin B(s_i, R)$.
Since, $X_i \subseteq B(s_i, R) \implies t_i \notin X_i$

If $s_i \in X_j$. The set X_j is defined by $s_j \implies \delta(s_j, s_i) < R < \frac{1}{2}$.
All vertices in X_j are within distance $< R$ of s_j.
By triangle inequality, any vertex in X_j is within distance $< 2R < 1$ from s_i.
Since $\delta(s_i, t_i) \geq 1$, we have $t_i \notin X_j$. $\square$

Estimating Probability of $e \in C$

Claim

$Pr(e \in C) \leq 2H_k x_e$, where H_k is the k-th Harmonic number, and it equals $H_k = \sum_{i=1}^{k} \frac{1}{i} \approx \ln k$.

Given the Claim, it is easy to see that the expected cost of the cut C will be within a factor of $O(\log k)$ of an optimal cut that separates k terminal pairs.

$$\begin{aligned} E[cost(C)] &= E\left[\sum_{e \in C} w(e)\right] \\ &= \sum_{e \in E} w(e) Pr(e \in C) \leq \sum_{e \in E} 2H_k w_e x_e = 2H_k z^* \end{aligned}$$

Theorem

Multicuts in a graph can be approximated within a factor of $O(\log k)$ in polynomial time that separates k- terminal pairs.

Proof of Claim

Claim

$Pr(e \in C) \leq 2H_k x_e$, where H_k is the k-th Harmonic number.

- Let $e = (u, v)$.

- We will consider distance from $s_1, \ldots, s_k$ to e.

- We define the distance from s_i to $e = (u, v)$ as $d(s_i, e) = \min(\delta(s_i, u), \delta(s_i, v))$.

- WLOG, assume that the order of vertices according to increasing distance from e be $s_1, s_2, \ldots, s_k$.
- In the random ordering of vertices in $s_1, \ldots, s_k$, consider when an end point u or v of e gets marked for the first time. Say it is u, and it gets marked by s_i.

- $u \in X_i$ and assume $\delta(s_i, u) \leq \delta(s_i, v)$.

- We have two cases (a) $v \in X_i$, (b) $v \notin X_i$.

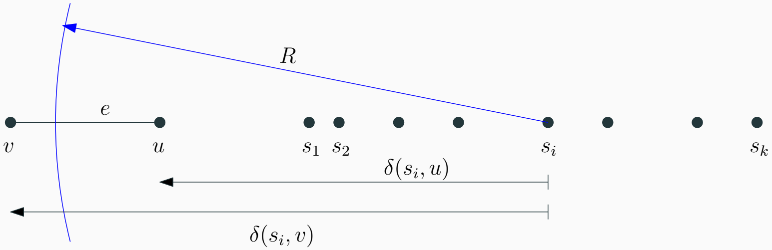
Proof Sketch of Claim (contd.)

Case (a): $v \in X_i$, i.e. v is also marked by s_i. Since both the ends of the edge $e = (uv)$ are in $X_i \implies e \notin C$.

Case (b): $v \notin X_i$. In this case $e \in C$, and we say s_i *cuts* e.
We want to estimate $Pr(s_i \text{ cuts } e)$.
Observe that s_i cuts e because of the following:

1. s_i marked u but not v.
2. $d(s_1, e) \leq d(s_2, e) \leq \cdots \leq d(s_k, e)$.
3. Among all the vertices $\{s_1, \ldots, s_k\}$, s_i is the first vertex that marked any of the end-points of e.
4. In the random order, none of the vertices that have smaller distance to e than s_i appeared. Otherwise, s_i won't be the first vertex marking an end of e.

An Illustration



Proof Sketch of Claim (contd.)

1. What is the probability that s_i comes before $s_1, \ldots, s_{i-1}$ in a random permutation?
   Answer: $\frac{1}{i}$.
2. What is the probability that s_i cuts e, given that s_i comes before $s_1, \ldots, s_{i-1}$?
   Answer: The radius $R \in (0, 1/2)$ should fall in the range $\delta(s_i, u) < R < \delta(s_i, v)$.
   Thus, the probability is $\leq \frac{\delta(s_i,v)-\delta(s_i,u)}{1/2} \leq \frac{x_e}{1/2} = 2x_e$.
3. What is the probability that s_i cuts e?
   Answer: $\frac{1}{i}2x_e$.
4. What is the probability that e is cut by any of $s_1, \ldots, s_k$?
   Answer: $\leq \sum_{i=1}^{k} Pr(s_i \text{ cuts } e) = \sum_{i=1}^{k} \frac{1}{i}2x_e = 2H_k x_e$.

□

References

References

1. Garg, Vazirani and Yannakakis, Multiway cuts in node weighted graphs, J. Algorithms 50(1): 49–61, 2004.
2. Garg, Vazirani and Yannakakis, Primal-dual approximation algorithms for integral flow and multicut in trees, Algorithmica 18(1): 3-20, 1997.
3. Calinescu, Karloff and Rabani, Approximation algorithms for the 0-extension problem, ACM-SIAM SODA 2001.
4. Fakcharoenphol, Rao and Talwar, A tight bound on approximating arbitrary metrics by tree metrics, JCSS 69(3): 485–497, 2004.
5. Williamson and Shmoys, The design of approximation algorithms, Cambridge University Press, 2011.
6. Several lecture notes (Anupam Gupta, Dinitz, Ravi).